NEW YORK UNIVERSITY
COURANT INSTITUTE – LIBRARY
4 Washington Place, New York 3, N. Y.

AEC Computing and Applied Mathematics Center

AEC RESEARCH AND DEVELOPMENT REPORT

TID-4500
20th Ed.

NYO-10,429

THE DEVELOPMENT AND USE

OF METHODS OF LR TYPE

by

Beresford Parlett

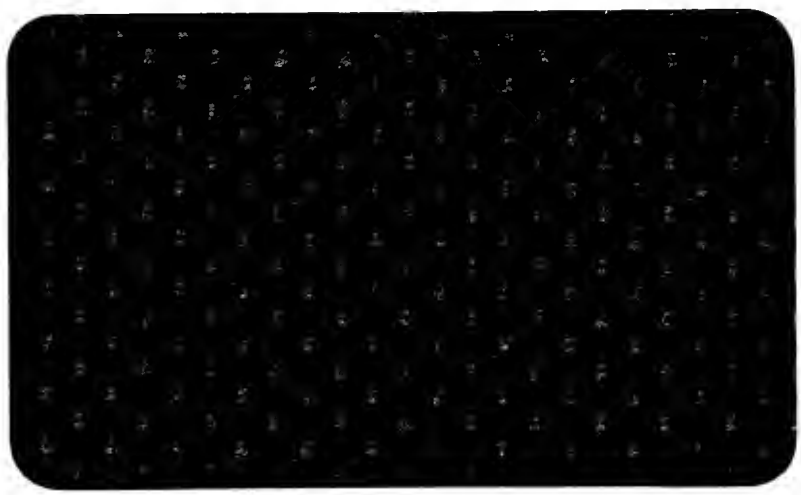
June 1, 1963

Courant Institute of Mathematical Sciences

NEW YORK UNIVERSITY
NEW YORK, NEW YORK

NEW YORK UNIVERSITY
COURANT INSTITUTE – LIBRARY
4 Washington Place, New York 3, N. Y.

NYO-10,429
c. 1



This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

A. Makes any warranty or representation, express or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of ony information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

AEC Computing and Applied Mathematics Center
Courant Institute of Mathematical Sciences
New York University

TID-4500
20th Ed.

NYO-10,429

THE DEVELOPMENT AND USE OF METHODS OF LR TYPE

by

Beresford Parlett

June 1, 1963

Contract No. AT(30-1)-1480

THE DEVELOPMENT AND USE OF METHODS OF LR TYPE

ABSTRACT

Given a non-singular matrix then factorize it into the product of two matrices. Form a new matrix by multiplying the two factors together in reverse order. Repeat the process and so produce a sequence of matrices all similar to the given matrix. This is the essense of an LR method and for several factorization techniques the sequence converges, quite generally, to a triangular matrix from which the eigenvalues may be read off.

The LR transformations discussed here preserve the form of a Hessenberg matrix ($a_{ij} = 0$ if $i > j+1$) and, for such matrices, the work involved in the transformation from one $N \times N$ matrix to the next is proportional to N^2 as against N^3 for full matrices. Using this and other devices the latest versions of the transformation are among the best available eigenvalue methods for general matrices.

The evolution of these algorithms can be traced back to classical function theory, each method being more feasible numerically than its predecessor.

TABLE OF CONTENTS

THE DEVELOPMENT AND USE OF METHODS OF LR TYPE

by

Beresford Parlett

1. Introduction

The rapid proliferation of numerical methods in linear algebra has made the family tree of their development almost unreadable. Therefore it is an unexpected pleasure to find a section of the tree whose evolution can be viewed, without too much distortion, as a simple sequential process, each development removing a defect of the previous method. This paper is written from the point of view of someone who seeks a "good" computational way of finding all the eigenvalues of a general matrix. This viewpoint has shaped the whole article and caused the exclusion of interesting but irrelevant facets of some of the topics.

The aim of the paper is to present the ideas and characteristics of the various algorithms in relation to each other so that the subject can be seen as a whole. Ample references are supplied to sources which give full details.

One way to find the eigenvalues of a matrix is to generate from it a sequence of similar matrices whose limit is triangular. The methods considered here are of this type. Each term is factorized into the product of two matrices. The next term is then the product of these factors in reverse order. In addition these

methods can be regarded as developments of the QD algorithm which itself can be seen as a practical way of using Hadamard's characterization of the poles of a rational function in terms of the coefficients of its Taylor series about a regular point.

The sections on QD are based on Henrici's interesting article [10], though the viewpoint is different and motivation is given for the introduction of certain essential quantities. Rutishauser's LR transformation [14] is suggested by an aspect of his QD algorithm. The QR transformation of Francis [6] avoids the possible "instability" of the original LR transformation but is slower. Wilkinson uses the LR method with row and column interchanges to produce an algorithm which is stable and requires no more arithmetic than the basic LR. In practice this variation seems both fast and accurate but no convergence proof has been published yet nor is it apparent how to adapt the technique to give, economically, the complex eigenvalues of real matrices.

The final section describes recent work of Ortega and others to produce a rival to the Sturm sequence method of finding the eigenvalues of real symmetric tridiagonal matrices.

2. Classical Theory

In order to find the eigenvalues of a matrix A it is completely satisfactory to replace A by $A + sI$ (I is the identity) where s is any convenient number. Hence there is no loss of generality in considering only non-singular matrices. Unless otherwise stated A is to be taken as a complex $N \times N$ matrix with eigenvalues λ_i ordered by the relation

$$|\lambda_1| \geq |\lambda_2| \geq \dots \geq |\lambda_N| > 0 .$$

The starting point for the investigations presented here is the casting of the eigenvalue problem in a function theoretic form which is going to prove most fruitful. Let x^* denote the conjugate transpose of the N dimensional vector x . For (almost) arbitrary vectors x and y we choose to consider a certain rational function of a complex variable z , regular at 0 and ∞ , which is defined by

$$f(z) = y^*(I-zA)^{-1}x .$$

The eigenvalues of A are the reciprocals of the poles z_i of f . Moreover since f is regular at the origin it is fully determined by its Taylor expansion there (which is convergent for $|z| < |\lambda_1|^{-1}$),

$$f(z) = \sum_0^\infty a_n z^n , \quad a_n = y^*A^n x , \quad A^0 = I .$$

In theory the a_n are easily calculated from A and so may be taken as known. The problem is now a special case of the problem considered and solved, in principle, by Hadamard at the end of the nineteenth century [9].

Determine the singularities of a meromorphic function from the coefficients of its Taylor series about a regular point.

In 1884 König proved that, if the only singularity on the circle of convergence of f is a simple pole at z_1 then $(a_n/a_{n+1}) \longrightarrow z_1$ as $n \longrightarrow \infty$. Hadamard [9] solved the general problem in 1892 with the aid of certain determinants, now usually called Hankel determinants, defined as follows

$$H_o^n = 1, \quad H_k^n = \det \begin{bmatrix} a_n & a_{n+1} & \cdots & a_{n+k-1} \\ a_{n+1} & a_{n+2} & \cdots & a_{n+k} \\ . & . & & . \\ . & . & & . \\ a_{n+k-1} & a_{n+k} & \cdots & a_{n+2k-2} \end{bmatrix} \quad (n = 0,1,2,\ldots : \; k = 1,2,\ldots) .$$

The basic result is contained in the following

LEMMA 1. Let $f(z) = \sum_0^\infty a_n z^n = f_o(z) + r(z)$, where f_o is regular analytic in $|z| \leq L^{-1}$ and r is rational with poles λ_i^{-1} satisfying

$$|\lambda_1| \geq |\lambda_2| \geq \ldots \geq |\lambda_m| > L > 0$$

(with the convention that a pole of order p is regarded as p concurrent simple poles) then as $n \longrightarrow \infty$

$$H_m^n = C(\lambda_1 \ldots \lambda_m)^n \{1 + O((L/\lambda_m)^n)\} ,$$

where C is a non-zero constant independent of n.

This is proved by Henrici in [10] for the case in which all the λ_i are distinct. The general case is covered by Golomb in [8].

The essence of the proof lies in the fact that a_n is of the form

$$a_n = B_1\lambda_1^n + \ldots + B_N\lambda_N^n + B_o$$

and so to find $\lambda_1\lambda_2\ldots\lambda_m$ it is only necessary to find some function of the $a_n, a_{n+1}, \ldots$ which is of the form

$$(\lambda_1 \ldots \lambda_m)^n F$$

where F may depend on anything but n. It turns out that H_m^n is just such a function.

From the lemma follows

COROLLARY 1. For large enough n, $H_m^n \neq 0$.

COROLLARY 2. As $n \longrightarrow \infty$, $\left(H_m^{n+1}/H_m^n\right) \longrightarrow \lambda_1\ldots\lambda_m$.

So classical function theory gives the solution that the k th eigenvalue of A is given by

$$\lambda_k = \lim_{n\to\infty} \left(H_k^{n+1}H_{k-1}^n \Big/ H_k^n H_{k-1}^{n+1}\right).$$

As Henrici points out this is unsatisfactory from the numerical point of view as there is no simple way to compute H_k^n. Aitken overcame this defect very neatly and produced a feasible numerical procedure. Rutishauser, with the QD algorithm, showed how to by-pass the computation of the Hankel determinants altogether with a resulting reduction in the amount of computation.

3. Aitken's Scheme

During the 1920's Aitken worked on the problem of finding all the zeros of a polynomial by generalising Bernoulli's method [1] and was led to study the Hankel determinants. He discovered a relation which allows the recursive calculation of the H_k^n from the a_i . This is most easily shown by arranging the H_k^n in the following table:

$$
\begin{array}{ccccccc}
1 & & & & & & \\
1 & H_1^0 & & & & & \\
1 & H_1^1 & H_2^0 & & & & \\
1 & H_1^2 & H_2^1 & H_3^0 & & & \\
1 & H_1^3 & H_2^2 & H_3^1 & H_4^0 & & \\
1 & H_1^4 & H_2^3 & H_3^2 & H_4^1 & H_5^0 & \\
. & . & . & . & . & . &
\end{array}
\qquad \text{(H - table)}
$$

Note that $H_1^n = a_n$ and so the first two columns are known.

In the case under consideration $f(z)$ has only N poles and so the H-table has only $N+1$ columns. This follows from a classical result (see Bieberbach [3], p. 319).

LEMMA 2. *Let* $f(z) = \sum a_n z^n = (b_o+\ldots+b_p z^p)/(c_o+\ldots+c_q z^q)$ *be a rational function* ($c_o \neq 0$, $c_q \neq 0$) . *Then*

$$H_{q+1}^n = 0 , \qquad n \geq \max (0, p+1-q) .$$

Conversely let p *be an integer such that* $H_q^n \neq 0$, $H_{q+1}^n = 0$ *for* $n \geq p+1-q$. *Then* f *is of the above form.*

In the present case the first half of the lemma is just a consequence of the Cayley-Hamilton theorem which yields

$$A^N = -\sum_{0}^{N-1} d_i A^i$$

where the d_i are the coefficients of the characteristic polynomial of A. Hence for $k = 0,1,2,\ldots$

$$a_{N+k} = y^*A^{N+k}x = -y^*(\sum_{i=0}^{N-1} d_i A^{i+k})x = -\sum_{i=0}^{N-1} d_i a_{i+k} .$$

Thus for $n = 0,1,2,\ldots$ the last column of H_{N+1}^k is a linear combination of the previous columns. Therefore $H_{N+1}^k = 0$.

The ratios of successive elements in the k th column of the H-table converge to the product of the largest k eigenvalues by Corollary 2 of Lemma 1, provided that $|\lambda_k| > |\lambda_{k+1}|$. By Corollary 1 the H-table ultimately exists, i.e. $H_k^n \neq 0$, $k \leq N$, n large enough. However it is possible for an early term to vanish and this would prevent the recursive computation of the table from previous terms. This possibility will be ignored here.

The relation between neighboring Hankel determinants

The relation which Aitken found for himself, [2], but which had been known to Hadamard, is

$$(H_k^n)^2 - H_k^{n-1}H_k^{n+1} + H_{k+1}^{n-1}H_{k-1}^{n+1} = 0 .$$

This is best remembered by taking any H_k^n as P and labelling its nearest neighbors in the H-table topographically as N, S, W, and

E . With such a stencil the relation above can be written as

$$P^2 = NS - WE . \qquad (R)$$

This formula can be used in two different ways:

(a) Start with the first two columns and work from left to right in the H-table, adding one column at a time by repeated use of $E = (NS-P^2)/W$.

(b) If the first two diagonals are known one can compute the lower diagonals successively working along the new diagonal from left to right, using $S = (P^2-WE)/N$.

Now (b) has the difficulty of finding the first two diagonals and, although he showed how to do this for a polynomial whose coefficients are given, Aitken in [2] favoured method (a).

For the computation of eigenvalues the generation of the H-table by (a) or (b), though feasible, still leaves a lot to be desired. With (a) it is necessary to compute $a_n = H_1^n = y^*A^n x$ for as many n as are necessary to obtain convergence of H_1^{n+1}/H_1^n to the desired degree of accuracy. With (b) there is the necessity of computing a_i for $i = 0,1,\ldots,2N-2$ and then the determinants H_k^0 and H_k^1 for $k = 1,2,\ldots,N$. If Gaussian elimination were used then, whilst calculating H_N^0 and H_N^1 , all the other H_i^0 and H_i^1 $(i = 1,2,\ldots,N-1)$, being leading principal minors, could be found. This technique, however, would not permit the use of interchanges in the Gaussian elimination and so extra precision would be required to ensure adequate accuracy.* Thus both (a) and

---

* In all $2N^3$ multiplication for the a_i and $2/3\ N^3$ double precision multiplications for the H_i^0 and H_i^1 .

(b) demand a formidable amount of computation before beginning to find the remainder of the H-table.

It was Rutishauser's achievement to change this scheme into a related, but far from obvious, form which is numerically preferable.

4. The QD Algorithm

What changes could be made in the H-scheme which might reduce the amount of initial computation? There is one natural observation: the H_k^n themselves are of less interest than the

$$q_k^n = \frac{H_k^{n+1} H_{k-1}^n}{H_k^n H_{k-1}^{n+1}} ,$$

since, as stated in section 2, for $|\lambda_1| > |\lambda_2| > \ldots > |\lambda_N|$. as $n \longrightarrow \infty$

$$q_k^n \longrightarrow \lambda_k .$$

As Henrici says in [10]: "It is remarkable that in the computation of the q_k^n . the determinants H_k^n do not have to be used if a set of auxiliary quantities is introduced." Their introduction is presented as a "fait accompli" by Rutishauser in [15]. Certainly this set of quantities arises naturally in looking at the continued fraction expansion of $f(z)$ (see [10], p. 36) but this author feels that a motivation for the choice of the auxiliary quantities, called e_k^n , can be made in the present context.

The key is to recast the basic recursion relation (R) so that it shows explicitly the connection between q_k^n and q_k^{n+1} . To do this clearly consider a typical portion of the H-table obtained by placing the following stencil on the table so that D is on H_k^n .

$$
\begin{array}{cccccc}
. & . & . & . & . & . \\
. & . & A & . & . & . \\
. & B & C & D & . & . \\
. & . & E & F & G & . \\
. & . & . & H & I & . \\
. & . & . & . & . & .
\end{array}
$$

Then

$$q_k^n = \frac{AF}{CD} \quad , \qquad q_k^{n+1} = \frac{CH}{EF} \quad .$$

To include all these elements two applications of (R) are needed, centred at C and F respectively, namely

$$C^2 = AE - BD \quad , \qquad F^2 = DH - EG \quad .$$

Now turn the first terms on the right hand sides into q_k^n and q_k^{n+1} respectively, obtaining

$$\frac{CF}{DE} = \frac{AF}{CD} - \frac{BF}{CE} \quad , \qquad \frac{CF}{DE} = \frac{CH}{EF} - \frac{CG}{DF} \quad . \tag{RR}$$

Reference to the stencil shows that BF/CE and CG/DF are of the same form and naturally ask to be named. If CG/DF $(= H_{k-1}^{n+1} H_{k+1}^{n} / H_k^n H_k^{n+1})$ is called e_k^n then BF/CE will be e_{k-1}^{n+1} . Transposing the second terms on the right hand sides of (RR) gives the following simple relation between the q and the e .

$$q_k^n + e_k^n = q_k^{n+1} + e_{k-1}^{n+1} \quad . \tag{R_1}$$

Another relation between the q and the e is required to

determine all the q_k^n and e_k^n from the q_1^n $(= a_{n+1}/a_n)$. It follows directly from the definitions of the q and e . A simple derivation comes from observing that

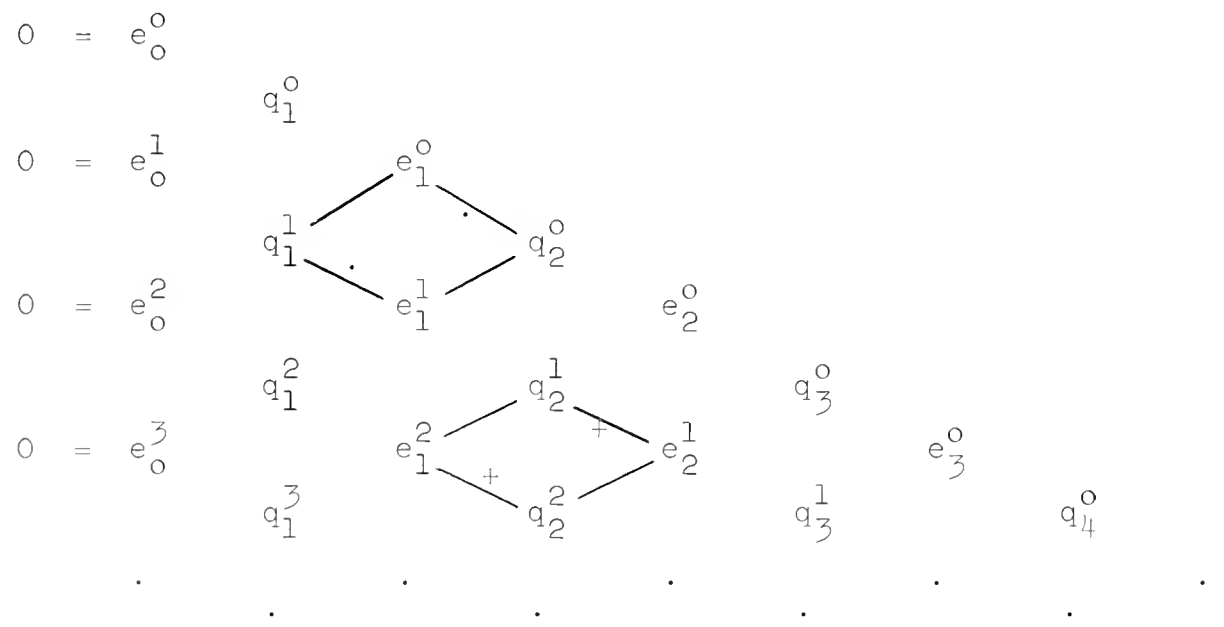
$$q_k^n \cdot e_k^n = \frac{AF}{CD}\cdot\frac{CG}{DF} = \frac{AG}{D^2} \; ; \quad q_k^{n+1} \cdot e_{k-1}^{n+1} = \frac{CH}{EF}\cdot\frac{BF}{CE} = \frac{BH}{E^2} \; .$$

Reference to the stencil shows that the right hand terms are of the same form but located in different diagonals, namely n and n+2 . By changing n and k so that both terms become CI/F^2 (in diagonal n+1) gives

$$q_k^{n+1} \cdot e_k^{n+1} = q_{k+1}^n \cdot e_k^n \; . \qquad (R_2)$$

The QD Scheme

Formula (R_1) shows that e_k^n is the difference of the quotients q_k^{n+1} and q_k^n modified by e_{k-1}^{n+1} . This indicates a little why Rutishauser named the following triangular table of q's and e's the Quotient-Difference Scheme.



Typical cases of (R_1) and (R_2) are shown pictorially in the table and explain why E. Stiefel named them the rhombus rules. (R_1) yields rhombi centred on q-columns, (R_2) yields rhombi centred on e-columns.

By alternate application of the rules the table may be generated, in theory, from either

(a) the first q-column moving right, or

(b) the first diagonal moving down.

Now, however, method (a) is quite impractical since rule (R_1) shows that

$$e_1^n = q_1^{n+1} - q_1^n .$$

For $|\lambda_1| > |\lambda_2|$, $q_1^n \longrightarrow \lambda_1$, as $n \longrightarrow \infty$. As n increases e_1^n becomes the difference of two almost equal numbers and, for computation with a fixed number of digits for each number, it will have fewer and fewer significant figures. Thus (e_1^{n+1}/e_1^n) and hence q_2^n will have little or no accuracy. Briefly, one says that method (a) is "unstable." Method (b), on the other hand, uses (R_1) in the form

$$q_k^{n+1} = q_k^n + e_{k+1}^n - e_{k-1}^{n+1}$$

which is ultimately completely safe when e_{k+1}^n and $e_{k-1}^{n+1} \longrightarrow 0$.

Difficulties in calculating the QD scheme

The danger in method (b) is that in the early stages, for small n, large values of e_{k-1}^{n+1} may occur which will cause loss of significant digits in the q_k^{n+1}. However with the QD scheme there is no

alternative but to use (b), the so-called progressive method. Rutishauser in [13] has analysed the effect of one small q_k^{n+1} and the resulting large value of e_k^{n+1} (in using (R_2)). He observed that only negative powers of e_k^{n+1} occur in diagonal $n+3$ and below. Thus the disturbance affects only the next two diagonals and he proposed some auxiliary modifications to be made in calculating some of the elements in diagonals $n+1$ and $n+2$. For diagonal $n+3$ and below the straightforward application of the rhombus rules is again sufficient. These precautions however are difficult to incorporate into an automatic program which must allow for the occurrence of several large e values in different columns of successive diagonals.

This same danger will appear again in the LR transformation and recent developments can be regarded as techniques to overcome this difficulty. See also the discussion by Wilkinson [18].

Although as $n \longrightarrow \infty$. $q_k^n \not\longrightarrow 0$ by assumption, it is still possible to have $q_k^n = 0$ for some small (finite) value of n. If this happens the QD scheme cannot be completed. Formally the scheme is said not to exist. This possibility will be ignored.

Calculation of the first diagonal

At this point it is worth observing that

$$q_1^n = H_1^{n+1}/H_1^n = y^*A^{n-1}x/y^*A^n x$$

and so the descent of column 1 of the QD scheme corresponds to the well known power method [17].

How can the first diagonal of the QD scheme be found indirectly? The answer is, surprisingly, by use of Lanczos' method of "minimized iterations", [11], with initial vectors x and y (from the definition of $f(z)$ in section 2). Thus, as Henrici points out, the QD scheme links the power method to the method of Lanczos.

The reader is referred to [11] and [10] for proofs and details of the algorithm. It produces a tridiagonal matrix similar to A and of the form (for $N = 5$)

$$J = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ 1 & \alpha_2 & \beta_2 & & \\ & 1 & \alpha_3 & \beta_3 & \\ & & 1 & \alpha_4 & \beta_4 \\ & & & 1 & \alpha_5 \end{bmatrix} .$$

By the LDU theorem ([4], p. 20), J may be written as the product LR where

$$L = \begin{bmatrix} q_1 & & & & \\ 1 & q_2 & & & \\ & 1 & q_3 & & \\ & & 1 & q_4 & \\ & & & 1 & q_5 \end{bmatrix} , \qquad R = \begin{bmatrix} 1 & e_1 & & & \\ & 1 & e_2 & & \\ & & 1 & e_3 & \\ & & & 1 & e_4 \\ & & & & 1 \end{bmatrix} .$$

It is easily verified that for $k = 1,2,\ldots,N$.

$$\alpha_k = q_k + e_{k-1} , \qquad e_0 = 0 .$$

$$\beta_k = q_k e_k , \qquad e_N = 0 .$$

Henrici shows ([10], p.33) that these q_k and e_k are the elements of the first diagonal of the QD scheme. In fact he shows that the n th diagonal can be obtained in the same way by using Lanczos' method with initial vectors $A^n x$ and y. In particular

$$\alpha_1^n = y^* A^{n+1} x / y^* A^n x = q_1^n$$

as it should.

Wilkinson [18] and Strachey and Francis [16] have examined the numerical reduction of a full matrix A to tridiagonal form J. The recommended method is to reduce A to an intermediate (Hessenberg) form H, indicated here for $N = 5$.

$$H = \begin{bmatrix} x & x & 0 & 0 & 0 \\ x & x & x & 0 & 0 \\ x & x & x & x & 0 \\ x & x & x & x & x \\ x & x & x & x & x \end{bmatrix} .$$

This reduction may be performed safely and accurately in single precision arithmetic by elimination methods if row and column interchanges are used. It requires $5/6\ N^3$ multiplications. The reduction of H to J requires $1/6\ N^3$ multiplications but is "unstable" and should be done in double precision. Comparison with section 3 shows that the first diagonal of the QD scheme can be obtained with approximately $1/3$ the computation required for the first two diagonals of the H-table.

When $|\lambda_1| > |\lambda_2| > \ldots > |\lambda_N|$ then $q_k^n \longrightarrow \lambda_k$, $e_k^n \longrightarrow 0$ as $n \longrightarrow \infty$. When there are eigenvalues with the same modulus then

the corresponding q and e columns may not converge. Fortunately this does not impair the usefulness of the algorithm. Discussion of what to do in this case is most simply presented in connection with the LR transformation and will appear in the next section.

5. The LR Transformation

In section 4 it was shown that the elements q_i^o and e_i^o of the first diagonal of the QD scheme are obtained from the LR decomposition of a certain tridiagonal matrix J, now to be called J_o, whose non-zero elements in row i are $1, \alpha_i, \beta_i$. This implies that, for $k = 1,2,\ldots,N$

$$\begin{aligned} q_k^o + e_{k-1}^o &= \alpha_k^o , & e_o^o &= 0 , \\ q_k^o e_k^o &= \beta_k^o , & \beta_N^o &= 0 . \end{aligned} \tag{5.1}$$

The rhombus rules. (R_1) and (R_2), which yield the next diagonal have a very interesting interpretation. Recall from the previous section that

$$L_o = \begin{bmatrix} q_1^o & & & & \\ 1 & q_2^o & & & \\ & 1 & q_3^o & & \\ & & \cdot & \cdot & \\ & & & & q_N^o \end{bmatrix} . \qquad R_o = \begin{bmatrix} 1 & e_1^o & & & \\ & 1 & e_2^o & & \\ & & 1 & e_3^o & \\ & & & \cdot & \cdot \\ & & & & 1 \end{bmatrix} .$$

From (R_1) and (R_2) and the rules of matrix multiplication follow the result

$$\begin{aligned} q_k^1 + e_{k-1}^1 &= q_k^o + e_k^o = (R_o L_o)_{kk} = \alpha_k^1 \quad \text{(say)} , \\ q_k^1 e_k^1 &= q_{k+1}^o e_k^o = (R_o L_o)_{k,k+1} = \beta_k^1 \quad \text{(say)} . \end{aligned} \tag{5.2}$$

It is easily verified that $R_o L_o$ is a tridiagonal matrix of the same form as J_o and may be called J_1. Now comparison of (5.2)

with (5.1) shows that the second diagonal of the QD scheme (q_i^1, e_i^1) contains precisely the elements of the LR decomposition of R_oL_o $(= J_1)$. Thus

J_o is factorized into L_oR_o.

The factors are multiplied in reverse order to give $R_oL_o = J_1$.

J_1 is factorized into L_1R_1.

Similarly the third diagonal of the QD scheme contains the elements of the LR decomposition of R_1L_1 $(= J_2)$, and so on for all the diagonals. In fact, for each positive integer n, equations (5.1) with superscript n define the elements α_i^n, β_i^n of a tri-diagonal matrix J_n. The J_n may be read, as it were, "between the (diagonal) lines" of the QD scheme. Clearly $J_{n+1} = R_nJ_nR_n^{-1}$ and if the eigenvalues λ_i of J_o (or of the original matrix A) have distinct moduli then, as $n \longrightarrow \infty$

$$J_n \longrightarrow L_\infty \quad \text{(a triangular matrix with } q_k^\infty = \lambda_k \text{)}.$$

Thus the QD scheme implicitly defines a convergent sequence of tri-diagonal matrices similar to J_o.

The LR algorithm

This interpretation of the QD scheme suggests a similar algorithm for a full non-singular matrix A_1. By the LDU theorem ([4], p. 20) A_1 may be factorized into the product of a left triangular matrix L_1 and a right triangular matrix R_1 which has 1's on the diagonal. The LR transformation is defined by

Rutishauser in[14] as follows. For $k = 1,2,\ldots$

$$\begin{cases} \text{Factorize } A_k \text{ into } L_k R_k . \\ \text{Form } R_k L_k \text{ and call it } A_{k+1} . \end{cases}$$

Clearly $A_{k+1} = R_k A_k R_k^{-1}$ and it may be asked when this sequence $\{A_k\}$ converges to a triangular matrix.

The answer, see [14], is contained in the following

THEOREM. If $A_1 = U \operatorname{diag}(\lambda_1,\ldots,\lambda_N)U^{-1}$ and

(a) $|\lambda_1| > |\lambda_2| > \ldots > |\lambda_N|$,

(b) The leading principal minors of U and U^{-1} are non-zero,

then $\lim A_k = A_\infty$ exists and is lower triangular with $(A_\infty)_{ii} = \lambda_i$.

If the $|\lambda_i|$ are not distinct then, loosely speaking, A_k may be said to converge to block triangular form. More precisely suppose that $|\lambda_1| = |\lambda_2| = \ldots = |\lambda_p| > |\lambda_{p+1}|$. Strictly speaking A_∞ may not exist but, as $k \longrightarrow \infty$. A_k becomes reduced. The elements in the first p rows may not converge but the characteristic polynomial of the leading principal submatrix of order p does converge to the monic polynomial with roots $\lambda_1,\ldots,\lambda_p$. The complementary submatrix has eigenvalues which converge to $\lambda_{p+1},\ldots,\lambda_N$. If any of these eigenvalues have equal modulus then this submatrix will become reduced also. Thus as $k \longrightarrow \infty$ the submatrix blocks which become isolated along the diagonal correspond to groups of eigenvalues of equal modulus. In the QD scheme, analogously, when successive q and e columns fail to converge, it turns out that certain polynomials, formed from the q's and e's in those columns and in the n th diagonal, do converge as $n \longrightarrow \infty$.

In the important case of real matrices with complex conjugate pairs of eigenvalues. A_k may be expected to have along the diagonal. for large k . isolated $2X2$ real submatrices which yield the eigenvalues very conveniently. This was the reason for stating in the previous section that the occurrence of eigenvalues of equal modulus does not impair the QD algorithm.

Shift of origin with the LR transformation

Usually the elements of A_k above the diagonal. $a_{ij}^{(k)}$ $(i < j)$, approach zero like $(|\lambda_i|/|\lambda_j|)^k$ as $k \longrightarrow \infty$. Thus convergence is (geometric) linear and depends on the ratios $|\lambda_i|:|\lambda_j|$. It happens, quite frequently in practice. that $|\lambda_N/\lambda_{N-1}|$ is smaller than the other ratios of consecutive eigenvalues and the (N,N) element is the first to become isolated as k increases. When the remaining elements of column N of A_k are zero to working accuracy then the $(N.N)$ element will be a good approximation to λ_N . The LR transformation may then be applied to the matrix obtained by omitting row and column N . In this way the eigenvalues may be found in turn. the order of the matrix being reduced at each stop.

Provided that s is considerably closer to λ_N than to λ_{N-1} it is preferable to apply the transformation to A_k-sI rather than to A_k since its $(N.N)$ element will converge to λ_N-s like $(|\lambda_N-s|/|\lambda_{N-1}-s|)^k$ as $k \longrightarrow \infty$: an improved convergence rate. A frequent choice for s is the current $(N.N)$ element but. as Wilkinson points out in [22]. this can prevent convergence by making

the sequence A_k cycle. He advocates putting

$$s = A_{NN} + \frac{1}{2} A_{N-1,N}$$

at each step. This implementation of the basic algorithm is very important. in practice, since it changes the rate of convergence from linear to quadratic.

It might be objected that A-sI will be nearly singular and hence its LR factorization will be liable to loss of accuracy. However this decomposition requires that only the first N-1 leading principal minors should not vanish ([4]. p. 20). Thus the transformation is safe numerically even on $A-\lambda_N I$ provided that λ_{N-1} is not too close to λ_N. Yet even if λ_{N-1} and λ_N are both small the algorithm may still be used. Now the final rows and columns of the L and R factors will be large and inaccurate except for the (N,N) element. The new matrix A_{k+1} will have an ill-determined last row. The errors in this row will not impair the accuracy of elements in other rows in further iterations. When the (N,N) element is accepted as $\lambda_N - \sum s$ then this erroneous last row is discarded and the computation proceeds on the (N-1)×(N-1) principal submatrix. For a proper discussion of this same phenomenon in the QR algorithm see [6].

Properties of the LR transformation

The LR decomposition of a full matrix requires $\frac{1}{3} N^3$ multiplications and the formation of RL demands a further $\frac{1}{6} N^3$. Thus

the transformation is not suitable for full matrices. On the other hand it is well adapted to some important special classes of matrices which will now be defined. To facilitate the discussion the writer proposes the following

DEFINITION. If $a_{ij} = 0$ for $i-j > m$ and $j-i > n$ then the diagonal width of the matrix $[a_{ij}]$ is (m,n). A band matrix of width ℓ is a matrix whose diagonal width is (ℓ,ℓ).

Rutishauser [14] shows that the width of band matrices is preserved under the LR transformation and this is very useful. Actually a more general result holds.

LEMMA. The diagonal width of a matrix is preserved under the LR transformation.

In particular the Hessenberg form, diagonal with $(N,1)$ or $(1,N)$, is preserved and this is important in the application to general matrices. A full matrix may be reduced to similar Hessenberg form in a stable manner with single precision arithmetic in $\frac{5}{6} N^3$ multiplications, see Wilkinson [18]. The LR transformation may then be applied to this Hessenberg matrix for which each iteration requires N^2 multiplications as against $\frac{1}{2} N^3$ for the full matrix. This important application is not discussed in [14] but is exploited in later developments of the LR method.

There is a variant of the basic algorithm given by Rutishauser for use with positive definite symmetric matrices which has cubic convergence [15]. See also section 8.

In summary here are the main features of the LR transformation,

in this writer's opinion, judged as a general eigenvalue routine.

ADVANTAGES

1. The diagonal width of a matrix is preserved.
2. The rate of convergence can be made quadratic by judicious use of origin shifts (which are easily coded).
3. Convergence to one eigenvalue (usually the smallest) is accompanied by convergence of the remaining diagonal elements to the other eigenvalues.
4. The "deflation" technique of omitting the final row and column when an eigenvalue has been accepted in the (N,N) position is both safe and easy to program.

DISADVANTAGES

1. The work involved in one iteration on a full matrix is prohibitive. However reduction of the full matrix to a similar Hessenberg matrix is accurate and simple and then each iteration requires less work. The further reduction of a Hessenberg matrix to tridiagonal form is not without numerical difficulties (instability) and needs extra precision in the computation.
2. The method is "unstable." More precisely the LR decomposition of A is equivalent to performing Gaussian elimination on A without row interchanges. It is known (see [19]) that this technique can involve severe loss of

significant digits in the elements of L and R. This is the analogue of the numerical difficulties of the progressive calculation of the QD scheme by diagonals.

One way of avoiding this last serious defect is to employ a stable factorization of a matrix. This is what Francis does in the QR algorithm [6]. Another way is to perform the Gaussian elimination with interchanges. This leads to a new algorith of Wilkinson [22] for real matrices with real eigenvalues. A variant of Wilkinson's algorithm for use on real matrices with complex eigenvalues is being studied by the author. All three techniques preserve the Hessenberg form which is an important property for the practical feasibility of a method.

6. The QR Transformation

For any matrix A there exists a unitary matrix Q and a right triangular matrix R with real, non-negative, diagonal elements such that

$$A = QR .$$

Moreover if A is non-singular then Q is unique and this is the matrix formulation of the Gram-Schmidt orthogonalization of the columns of A. Francis in [6] uses this result to define and develop his QR transformation as follows: let $A = A_1$, then for $k = 1,2,\ldots$

$$\begin{cases} \text{Factorize } A_k \text{ into } Q_k R_k . \\ \text{Form } R_k Q_k \text{ and call it } A_{k+1} . \end{cases} \tag{6.1}$$

Now $A_{k+1} = Q_k^* A_k Q_k$ and so the $\{A_k\}$ form a sequence of unitarily similar matrices in contrast to the LR transformation which generates a sequence of merely similar matrices. It is generally believed that, in numerical work, unitary transformations are very stable. Francis cites Fike's result [5] that the condition of an eigenvalue is unchanged (i.e. not made worse) when the matrix undergoes an orthogonal congruence.

The basic property

Francis proves that if A is non-singular and has eigenvalues all of distinct moduli then, as $k \longrightarrow \infty$, the elements of A_k

below the principal diagonal tend to zero, the moduli of those above the diagonal converge, and the elements on the principal diagonal tend to the eigenvalues. If A has some eigenvalues of equal modulus then these will be the limits of the eigenvalues of certain principal submatrices of A_k whose elements need not converge.

In contrast to LR, the QR transformation only preserves the band width (ℓ,ℓ) of Hermitian matrices. Fortunately, however, the upper Hessenberg form (width $(N,1)$) is preserved. Thus a general matrix may be reduced to this form initially and the QR transformation may then be applied to the upper Hessenberg matrix with great saving of computation.

In common with the LR transformation the QR transformation usually yields the eigenvalues ordered in magnitude down the diagonal of A_∞ . In practice the smallest eigenvalue usually "appears" first, located in the (N,N) element. Thus the translation technique described in the discussion of the LR method may be used here in exactly the same way to obtain faster convergence. This generalized QR transformation is defined by Francis as follows. Let $A = A_1$, then for $k = 1,2,\ldots$

$$\begin{cases} \text{Factorize } A_k - s_k I \text{ into } Q_k R_k \ , \\ \text{Form } R_k Q_k + s_k I \text{ and call it } A_{k+1} \ . \end{cases} \tag{6.2}$$

The shifts s_i are assumed to be distinct from, but approaching, the smallest eigenvalue. There is some freedom for skill in their choice. Usually one inspects the roots of the bottom 2×2 principal

submatrix. If they differ markedly from the corresponding values at the previous iteration then $s_k = 0$. Otherwise s_k is taken as the smaller of the two roots in magnitude.

Note that $A_{k+1} = Q_k^* \ldots Q_1^* A_1 Q_1 \ldots Q_k$ and so each A_k is similar to A_1. It would, perhaps, be more natural to define each A_{k+1} as $R_k Q_k$ so that A_{k+1} would be similar to $A_1 - (s_1 + \ldots + s_k)I$. The only objection to this variant is that it would not allow a sophisticated trick which Francis uses on real matrices and which will be described below. Either technique may be used for complex matrices.

The formation of Q

Let A_1 be an upper Hessenberg matrix. By (6.2) Q_1 is the unitary matrix such that premultiplication of $A_1 - s_1 I$ by Q_1^* eliminates the subdiagonal elements. There are only N-1 of these and Q_1^* may conveniently be taken as the product of N-1 unitary matrices, the i th of which eliminates element (i+1,i). Each may be taken simply as the identity except for a 2×2 principal submatrix of the form

$$\begin{bmatrix} e^{i\alpha} \cos\theta & -e^{i\beta} \sin\theta \\ e^{i\gamma} \sin\theta & e^{i\delta} \cos\theta \end{bmatrix}, \quad \alpha+\beta+\gamma+\delta \equiv 0 \pmod{2\pi}, \ \theta \text{ real}. \qquad (6.3)$$

Such matrices are the unitary analogues of Jacobi's two dimensional rotation matrices. For real matrices $\alpha = \beta = \gamma = \delta = 0$.

Using these elementary unitary transformations one step of the

QR method requires approximately $4N^2$ multiplications as against N^2 multiplications for one step of the LR method, both being applied to a Hessenberg matrix. Stability has been achieved at the cost of a substantial increase in the amount of computation. Of all the algorithms considered so far in this paper the QR transformation is the first that can be used safely as a feasible general eigenvalue routine.

The double QR iteration

Now consider the important special case of real matrices with complex eigenvalues. In order to realize the faster rate of convergence of the generalized QR method (using shifts) on such matrices complex shifts must be employed at some stage. After that stage the matrices will contain complex elements and so will require all the extra work involved in treating complex matrices (double the storage, at least four times the work). There is thus strong incentive for seeking an algorithm which produces a real matrix from which the complex eigenvalues can be calculated easily as conjugate roots of 2×2 real principal submatrices.

Consider two steps of the generalized QR transformation of a general real matrix A_1. From (6.2) it follows that

$$A_3 = Q_2^* A_2 Q_2 = Q_2^* Q_1^* A_1 Q_1 Q_2 \qquad (6.4)$$

and also the important result that

$$\begin{aligned} Q_1 Q_2 R_2 R_1 &= Q_1 (A_2 - s_2 I) R_1 \\ &= (A_1 - s_2 I) Q_1 R_1 \\ &= (A_1 - s_2 I)(A_1 - s_1 I) \equiv G . \end{aligned} \qquad (6.5)$$

Thus. setting $Q_o = Q_1 Q_2$ and $R_o = R_2 R_1$, the formulas above say that $A_3 = Q_o^* A_1 Q_o$ where, by uniqueness, $Q_o R_o$ is the QR factorization of the non-singular matrix G . It is now clear what should be done. Suppose that s_1 is a complex shift but that s_2 is forced to be $\bar{s}_1$, then G will be real, Q_o will be its (real) orthogonal factor, A_3 will be real, and the computation of complex A_2 will be circumvented.

The solution is therefore to use the generalized QR transformation with the extra proviso that every complex shift is followed by its conjugate, the intermediate complex matrix being avoided as indicated above. What is the price for this convenience? Ignoring the storage of the auxiliary matrix G it turns out that the formation of G , its decomposition, and the formation of A_3 (one double step) require at least as much work as two real QR steps. Apart from the decided advantage of keeping the arithmetic real there is no extra gain.

Is there a better solution? For a full real matrix the answer is no. However it has already been stated that it is a safe economy to reduce a full matrix to Hessenberg form before using the QR transformation. Now Francis exploits a property of Hessenberg matrices to produce an algorithm in which a <u>double</u> QR step with complex conjugate shifts requires only $5N^2$ multiplications as against $4N^2$ for one basic real QR step. This is most satisfactory.

A property of Hessenberg matrices

The fact he uses is the following. If any matrix M is unitarily similar to an upper Hessenberg matrix H with real, non-negative, subdiagonal elements then, in general, the first column of the transformation matrix U uniquely determines the remaining columns of U and all of H. More precisely let

$$UH = MU \quad , \tag{6.6}$$

then this equation and the form of H yield a recursive procedure for calculating the U_i and the h_{ij} from U_1 and M. Subscripts on matrices denote columns. Let $j = 1,2,\ldots,N$ and at the j th stage take as known $H_1,\ldots,H_{j-1}$ and $U_1,\ldots,U_j$. The formulas for H_j and U_{j+1} come from equating the j th column of each side of (6.6). For $i = 1,\ldots,j$ the orthogonality of U_i to the other columns of U gives

$$h_{ij} = U_i^* MU_j \quad . \tag{6.7}$$

Using these values take all known quantities to the right side of $UH_j = MU_j$ to get

$$h_{j+1,j}U_{j+1} = MU_j - \sum_{i=1}^{j} h_{ij}U_i \equiv V_{j+1} \quad .$$

Since U_{j+1} must be of unit length, i.e. $\|U_{j+1}\| = 1$, it follows that

$$\left.\begin{aligned} h_{j+1,j} &= \|V_{j+1}\| \quad , \\ U_{j+1} &= \left(\frac{1}{h_{j,j+1}}\right) V_{j+1} \quad . \end{aligned}\right\} \tag{6.8}$$

The procedure comes to an end at the first j ($\leq N$) for which $h_{j+1,j}$ vanishes. Francis shows that in the way he chooses U_1 the process will not end prematurely and so U_1 does indeed determine the whole transformation.

Double QR on a Hessenberg matrix

The application of this fact is to the implication of (6.4) and (6.5). Take A_1 to be a real upper Hessenberg matrix (zeros below the subdiagonal) and recall that this form is preserved under the QR transformation. So the orthogonal transformation of A_1 into A_3 by Q_o is uniquely determined by the first column of Q_o. Now $Q_oR_o = G$ and R_o is right triangular. Therefore column 1 of Q_o is just column 1 of G normalized to unit length. Only the first column of G is needed to compute A_3 and this column contains at most 3 non-zero elements. The computation of G is avoided thus saving storage space and $1/12\ N^3$ multiplications per iteration.

The execution of the method is the following. Apply to A_1 any orthogonal congruence the matrix of which (say P_1) has as its first column the normalized first column of G. This will destroy the Hessenberg form. Now apply to $P_1^*A_1P_1$ any sequence of orthogonal congruences (by $P_2,\ldots,P_N$, say) such that

$$P_N^* \ldots P_1^*A_1P_1 \ldots P_N \qquad (6.9)$$

is restored to upper Hessenberg form with non-negative subdiagonal elements. The one proviso on the $P_2,\ldots,P_N$ is that the first column of $P_1\ldots P_N$ must be the first column of P_1. This is easily achieved. Finally, then, if no subdiagonal element vanishes the new

matrix must be $A_3 = Q_o^* A_1 Q_o$ since Q_o and P_1 have the same first column.

The recursion (6.7) and (6.8) will not end prematurely

It remains to show that no subdiagonal element of (6.9) will vanish. So far use has been made of the form of A_3 but not of the form of A_1. There is no loss of generality in assuming that A_1 has positive subdiagonal elements. (If a zero occurs then A_1 is reduced and the two principal submatrices may be taken separately.) Francis proves (Theorem 11) that this assumption ensures that no subdiagonal element of A_3 vanishes provided that s_1 is not an eigenvalue and that A_1 is diagonalizable (linear elementary divisors). However the following simple argument shows that this last assumption (diagonalizability) is not necessary.

Suppose that $UH = A_1 U$ where H is an upper Hessenberg matrix and $U_1 = cG_1$ with $1/c = \|GI_1\| \neq 0$. Here I_1 is the first column of I. If the recursion relation (6.7) and (6.8) ends prematurely then for some $j < N$

$$U_{j+1} = A_1 U_j - \sum_{i=1}^{j} h_{ij} U_i = 0 .$$

Hence there is a polynomial $p(z)$, of degree j, such that

$$U_{j+1} = p(A_1)U_1 = p(A_1)GI_1 = 0 .$$

This is so because each U_i, $i \leq j$, is by construction the

vector U_1 premultiplied by a polynomial in A_1 of degree i. Now G is also a polynomial in A_1 and is non-singular by hypothesis. So G commutes with $p(A_1)$ and, further, $Gp(A_1)I_1 = 0$ implies that $p(A_1)I_1 = 0$. Now A_1 has positive subdiagonal elements and so $p(A_1)I_1$ has a non-zero $(j+1)$ th element which comes from the highest degree term of p and cannot be affected by the contributions of the lower degree terms. This contradiction shows that this choice of U_1 does determine uniquely the whole transformation and so H must be A_3.

Francis describes in detail the execution of the algorithm and the choice of the P_i for which he uses matrices of the type used by Householder to reduce a symmetric matrix to tridiagonal form [20]. This double iteration is approximately four times faster than a straightforward computation of A_3 via complex A_2.

7. The LR Transformation with Interchanges

Wilkinson [22] has proposed a variant of the LR transformation which removes the instability and requires no extra arithmetic operations. This new technique does not preserve the diagonal width (see section 5) of a matrix in general but, fortunately, it does preserve the Hessenberg form. (diagonal width (N,1) or (1,N)) and only such matrices will be discussed in this section. The method was proposed for Hessenberg matrices which are either real with real eigenvalues or complex.

In section 5 it was shown that one iteration of the standard LR transformation can be seen as the result of premultiplying the current matrix A_k by a matrix L_k^{-1} to produce an upper (or right) triangular matrix R_k and then postmultiplying R_k by L_k to complete the similarity transformation. L_k itself is a product of elementary matrices (the identity with one non-zero subdiagonal element). In Wilkinson's variant L_k is replaced by a matrix L_k' which is not in general triangular, being again a product of elementary matrices but with an interchange matrix inserted between consecutive elementary matrices. These interchanges ensure that no element in the elementary matrices exceeds 1 in magnitude.

Shifts of origin are used in the same way as described in section 5 to increase the convergence rate. Wilkinson reports in [22] that this method is among the fastest and most accurate in his repertoire. All that remains is for a proof of convergence to appear in print.

The very fact that seems to prevent this also seems to prevent the imitation of the double iteration used by Francis (see section 6) for real matrices with complex eigenvalues.

How the interchanges affect the iteration

Straightforward Gaussian elimination used in solving systems of linear equations is equivalent to the LR factorization of the matrix of coefficients. It is well known, see [19], that disastrous errors can be introduced when the computation is carried out with numbers held to a fixed number of digits. The cure for this condition is to perform the Gaussian elimination "with interchanges." Before the elements below the main diagonal in a column are eliminated two rows are interchanged so that the diagonal element (or pivot) is not exceeded in magnitude by any element below it. This method gives rise to a new factorization of the coefficient matrix.

To see this in more detail consider a 5×5 upper Hessenberg matrix H_1. Let M_i denote a 5×5 matrix which is the identity except for a non-zero $(i+1,i)$ element. Let I_{ab} denote the identity matrix with rows a and b interchanged. Now the elimination indicated above may be written in matrix notation as

$$M_4 I_{4d} M_3 I_{3c} M_2 I_{2b} M_1 I_{1a} H_1 = R' .$$

where $a = 2$ or 1, $b = 3$ or 2, $c = 4$ or 3, $d = 5$ or 4 according as there was an interchange or not. When convenient this equation may be written in the following way

$$(M_4)(I_{4d}M_3I_{4d})(I_{4d}I_{3c}M_2I_{3c}I_{4d})(I_{4d}I_{3c}I_{2b}M_1I_{2b}I_{3c}I_{4d})P'^{-1}H_1 = R' ,$$

where $P'^{-1} = I_{4d}I_{3c}I_{2b}I_{1a}$. The matrices in parentheses, and hence their product L'^{-1}, are left triangular and so elimination with interchanges corresponds to the factorization

$$H_1 = P'L'R'$$

in contrast to the standard decomposition

$$H_1 = L_1R_1 .$$

The new iterate of H is given by

$$H^{(2)} = R'P'L' = (L')^{-1}(P')^{-1}H_1P'L'$$

in contrast to the basic LR iterate

$$H_2 = R_1L_1 = L_1^{-1}H_1L_1 .$$

An important property of the basic LR transformation is that

$$H_k = F_k^{-1}H_1F_k$$

where F_kS_k is the LR decomposition of $(H_1)^k$ and $F_k = L_1 \ldots L_k$, $S_k = R_k \ldots R_1$. It seems difficult to find a usable analogue of this property for the new algorithm. For example $H^{(2)}$ is factored into $P'L'R'$ and

$$H^{(3)} = R''P''L''$$

$$= (L'')^{-1}(P'')^{-1}H^{(2)}P''L''$$

$$= (L'')^{-1}(P'')^{-1}(L')^{-1}(P')^{-1}H_1P'L'P''L'' .$$

Also

$$(P'L'P''L'')(R''R') = P'L'H^{(2)}R' = (P'L'R')^2 = (H_1)^2,$$

but the two matrices on the left give neither the LR nor the PLR factorization of $(H_1)^2$. This prevents the straightforward imitation of the double QR transformation, as is shown by a comparison with the appropriate part of section 6 (formula 6.4).

For real matrices with real eigenvalues it would suffice to show that the interchanges must at some stage become unnecessary for stability because after that stage the iteration reduces to the basic LR transformation.

8. Symmetric Tridiagonal Matrices

A popular and excellent method of finding the eigenvalues of a full real symmetric matrix is to transform the matrix into tridiagonal symmetric form by a finite sequence of orthogonal transformations using either Givens' [7] or Householder's [20] method. The eigenvalues of the tridiagonal matrix are then computed successively by the Sturm Sequence technique [7] which has much to recommend it, especially if there are clusters of close eigenvalues. However there is an alternative which seems to be from three to six times faster than the Sturm Sequence algorithm but which is not so reliable in finding close or multiple eigenvalues. This alternative has been noted, among others, by Wilkinson, Ortega, Caffrey, and Kaiser.

For a positive definite real symmetric matrix A_1 there is a simple variant of the LR transformation which is stable and preserves symmetry. The Choleski decomposition of A_1 expresses A_1 as $L_1 L_1^T$, where L_1 is a left triangular matrix. Next A_2 is defined as $L_1^T L_1$ and then decomposed into $L_2 L_2^T$; $A_3 = L_2^T L_2$ and so on. Rutishauser [14] proved that under this transformation A_k converges to a diagonal matrix as $k \longrightarrow \infty$. This algorithm preserves the diagonal width of a matrix and so would seem a sure and possibly swift method of obtaining the eigenvalues when A_1 is tridiagonal. Let

$$A_1 = \begin{bmatrix} a_1 & b_1 & & & \\ b_1 & a_2 & b_2 & & \\ & b_2 & a_3 & b_3 & \\ & & b_3 & a_4 & b_4 \\ & & & b_4 & a_5 \end{bmatrix} \cdot \quad L_1 = \begin{bmatrix} d_1 & & & & \\ \ell_1 & d_2 & & & \\ & \ell_2 & d_3 & & \\ & & \ell_3 & d_4 & \\ & & & \ell_4 & d_5 \end{bmatrix} .$$

From $A_1 = L_1 L_1^T$ it follows that

$$d_i^2 = a_i - \ell_{i-1} \cdot \quad i = 1,\ldots,5, \quad \ell_o = 0 ,$$

$$\ell_i = b_i/d_i \quad . \quad i = 1,\ldots,4 .$$

Since A_1 is positive definite it follows that the $d_i > 0$ and so the square roots $\sqrt{a_i - \ell_{i-1}^2}$ are certainly real. However these N square roots would slow down the numerical execution of the algorithm to such an extent that it would not be competitive with the Sturm sequence method.

The modified LR algorithm

The speed of the recommended procedure derives from the fact that no square roots need be calculated. To see this consider the formation of A_2 from the decomposition of A_1 . Let

$$L_1^T L_1 = A_2 = \begin{bmatrix} \bar{a}_1 & \bar{b}_1 & & & \\ \bar{b}_1 & \bar{a}_2 & \bar{b}_2 & & \\ & \bar{b}_2 & \bar{a}_3 & \bar{b}_3 & \\ & & \bar{b}_3 & \bar{a}_4 & \bar{b}_4 \\ & & & \bar{b}_4 & \bar{a}_5 \end{bmatrix} ,$$

then

$$\bar{a}_i = d_i^2 + \ell_i^2 , \qquad i = 1,\ldots,N, \quad \ell_N = 0$$

$$\bar{b}_i = d_{i+1}\ell_i , \qquad i = 1,\ldots,N-1 .$$

The important observation is that it is not necessary to know ℓ_i and d_i to obtain A_2 but only ℓ_i^2 and d_i^2. Of course this means that A_2 is determined only up to the signs of the $\bar{b}_i$. However a change in the sign of some (or all) of the $\bar{b}_i$ produces a matrix which is similar to A_2 via a diagonal matrix with diagonal elements ± 1. Thus the ambiguity of sign may be ignored and A_2 may be considered as determined by the $\bar{a}_i$ and $\bar{b}_i^2$. The appropriate modification of the straightforward algorithm is just to square the equations giving ℓ_i and $\bar{b}_i$. The step from A_i to A_{i+1} then requires only $2N$ additions, $(N-1)$ multiplications, and $(N-1)$ divisions.

Stability

Much has been made in this paper of the instability of the numerical LR transformation for general matrices. It is therefore important that the Choleski decomposition of a positive definite symmetric matrix is stable. A thorough discussion of the matter is given by Wilkinson [21]. For tridiagonal matrices the equation

$$0 < d_i^2 = a_i - \ell_{i-1}$$

shows that each $|\ell_i|$ is bounded by $\max_i a_i$.

It is also apparent that the modified algorithm is applicable to symmetric matrices which are not positive definite. Neither convergence to diagonal form nor stability are guaranteed in this event but for an initially positive definite matrix A_1 this modification does allow the use of simple origin shifts (e.g. the (N,N) element) which may give A_k a negative eigenvalue for some k. However for positive definite matrices Rutishauser has a technique [15] for choosing the origin shifts in a more complicated manner which yields cubic convergence to the diagonal form. It is not always advisable to force any symmetric matrix to be positive definite by use of a large origin shift since this could cause the loss of significant digits in the small elements of the given matrix.

The modified QR algorithm

For symmetric tridiagonal matrices there is a technique whose stability is assured. The QR transformation is greatly simplified with such matrices and it can be modified so that no square roots need be taken. Each iteration then requires $5N$ additions, $3N$ multiplications, and $2N$ divisions. For a more detailed discussion see Ortega [12].

Acknowledgement

This paper came from a talk given by the author at the Courant Institute, NYU. The author wishes to thank S. Schechter for helpful advice on the presentation of some of the material.

REFERENCES

[1] A. C. Aitken, On Bernoulli's numerical solution of algebraic equations, Proc. Roy. Soc. Edinburgh 46, 289-305 (1925/26).

[2] A. C. Aitken, Further numerical studies in algebraic equations and matrices, Proc. Roy. Soc. Edinburgh 51, 80-90 (1931).

[3] L. Bieberbach, Lehrbuch der Funktionentheorie, vol. II, 2d ed. (Chelsea Publishing Co., New York, N.Y., 1945).

[4] V. N. Fadeeva, Computational Methods of Linear Algebra, Dover Publications (1959).

[5] C. T. Fike, Note on the practical computation of proper values, J. Assoc. Comput. Mach., 6, 360 (1959).

[6] J. G. F. Francis, The QR Transformation - a unitary analogue to the LR transformation, Parts 1 and 2, Comp. Journal, 4, 265-271 and 332-345, (1961/62).

[7] W. Givens, Numerical computation of the characteristic values of a real symmetric matrix, Report ORNL 1574, Oak Ridge National Laboratory, (1945).

[8] M. Golomb, Zeros and poles of functions defined by Taylor series, Bull. Am. Math. Soc., 49, 581-592 (1943).

[9] J. Hadamard, Essai sur l'étude des fonctions données par leur développement de Taylor. J. Math. Pures Appl. (4), 8, 101-186 (1892).

[10] P. Henrici, The Quotient-Difference Algorithm, NBS Appl. Math. Series No. 49, (1958).

[11] C. Lanczos, An iteration method for the solution of the eigenvalue problem of linear differential and integral operators, J. Research NBS, 45, 255-282 (1950) RP2133.

[12] J. Ortega, On the LL^T and QR methods for symmetric tridiagonal matrices, in preparation.

[13] H. Rutishauser, Der Quotienten-Differenzen-Algorithmus, Z. angew. Math. Physik, 5, 233-251 (1954).

[14] H. Rutishauser, Solution of Eigenvalue Problems with the LR transformation, NBS Appl. Math. Series No. 49, (1958).

[15] H. Rutishauser, Über eine kubisch konvergente variante der transformation, Z. angew. Math. Mech., 40, 49-54 (1960).

[16] C. Strachey and J. G. F. Francis, Reduction of a matrix to codiagonal form by elimination, Comp. Journal, 4, 168-176 (1961).

[17] J. H. Wilkinson, The use of iterative methods for finding the latent roots and vectors of matrices, Math. Tables and Other Aids to Comput., 9, 184-191 (1955).

[18] J. H. Wilkinson, Stability of the reduction of a matrix to almost triangular and triangular forms by elementary similarity transformations, J. Assoc. Comput. Mach., 6, 336-359 (1959).

[19] J. H. Wilkinson, Rounding errors in algebraic processes, Information Processing, Unesco, Paris, 44-53 (1960).

[20] J. H. Wilkinson, Householder's method for the solution of the algebraic eigenproblem, Comp. Journal, 3, 23-37 (1960).

[21] J. H. Wilkinson, Error analysis of direct methods of matrix inversion, J. Assoc. Comput. Mach., 8, 281-330 (1961).

[22] J. H. Wilkinson, Calculation of eigenvalues (assured real) of a lower Hessenberg matrix by a variant of the LR technique, in preparation.

AUG 1 1963

DATE DUE

AUG 11 '66

GAYLORD

PRINTED IN U.S.A.